dV/dt - Accelerating the Rate of Progress towards Extreme Scale Collaborative Science

Funded by DOE under the

Scientific Collaborations at Extreme-Scales Program

3 year project (Fall 2012)

- Lead Institution: University of Wisconsin - Madison
  Lead PI: Miron Livny

- Co-PIs:
- William Allcock, U-Chicago, Argonne National Laboratory
- Douglas Thain, University of Notre Dame
- Frank Wuerthwein, University of California–San Diego
- Ewa Deelman, University of Southern California

Thesis

- Researchers come together into dynamic collaborations and employ a number of applications, software tools, data sources, and instruments
- They have access to a growing variety of processing, storage and networking resources
- Goal: "make it easier for scientists to conduct large-scale computational tasks that use the power of computing resources they do not own to process data they did not collect with applications they did not develop"

Challenges today

- Estimate the application resource needs
- Find the appropriate computing resources
- Acquire those resources
- Deploy the applications and data on the resources
- Manage applications and resources during run

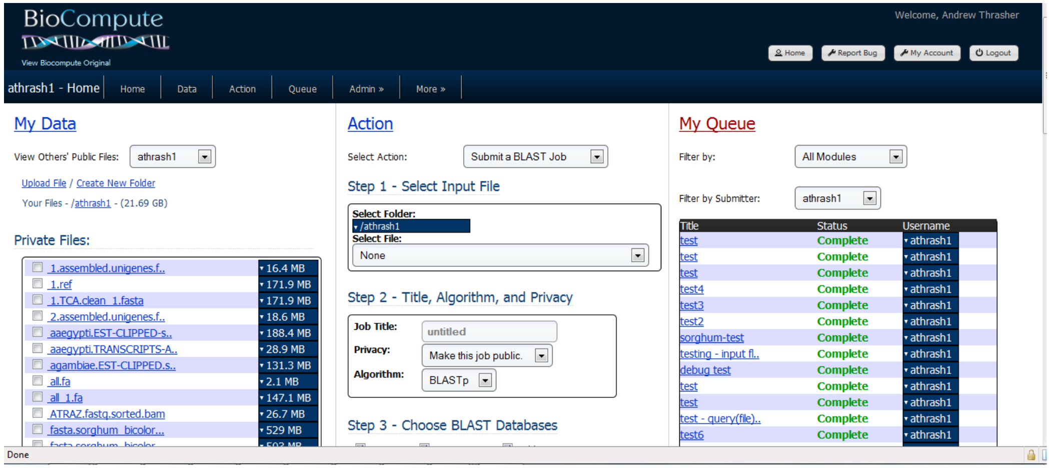
Approach

- A planning framework that covers the entire spectrum of computing resources—processing, storage, networking, and software
- The framework that encompasses the five phases of collaborative computing—estimate, find, acquire, deploy, and use

Experimental Foundation

- Real-world applications
- State of the art computing capabilities—ALCF and OSG
- Campus resources at ND, UCSD and UW
- Commercial cloud services
- Experimentation from the point of view of a collaboration member: "submit locally and compute globally"
- Pay attention to the cost involved in acquiring the resources and the human effort involved in software and data deployment and application management

Applications:
Portal Generated Workflows/ use Makeflow WMS



BioCompute

View Biocompute Original

Welcome, Andrew Thrasher

Home | Report Bug | My Account | Logout

athrash1 - Home | Home | Data | Action | Queue | Admin » | More »

My Data

View Others' Public Files: athrash1

Upload File / Create New Folder

Your Files - /athrash1 - (21.69 GB)

Private Files:

□ 1.assembled.unigenes.f..	16.4 MB
□ 1.ref	171.9 MB
□ 1.TCA.clean_1.fasta	171.9 MB
□ 2.assembled.unigenes.f..	18.6 MB
□ aaegypti.EST-CLIPPED-s..	188.4 MB
□ aaegypti.TRANSCRIPTS-A..	28.9 MB
□ agambiae.EST-CLIPPED.s..	131.3 MB
□ all.fa	2.1 MB
□ all_1.fa	147.1 MB
□ ATRAZ.fastq.sorted.bam	26.7 MB
□ fasta.sorghum_bicolor...	529 MB

Action

Select Action: Submit a BLAST Job

Step 1 - Select Input File

Select Folder: /athrash1

Select File: None

Step 2 - Title, Algorithm, and Privacy

Job Title: untitled

Privacy: Make this job public.

Algorithm: BLASTp

Step 3 - Choose BLAST Databases

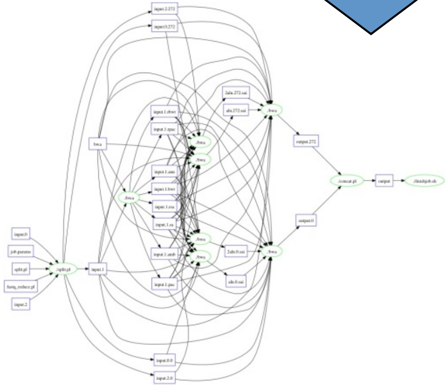
My Queue

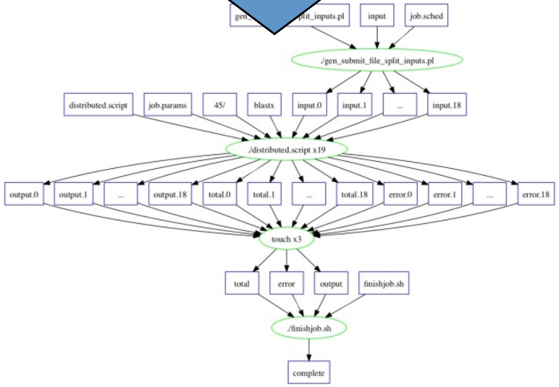
Filter by: All Modules

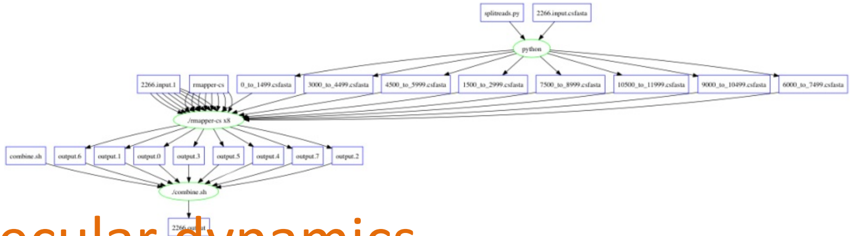
Filter by Submitter: athrash1

Title	Status	Username
test	Complete	athrash1
test	Complete	athrash1
test	Complete	athrash1
test4	Complete	athrash1
test3	Complete	athrash1
test2	Complete	athrash1
sorghum-test	Complete	athrash1
testing - input fl..	Complete	athrash1
debug test	Complete	athrash1
test	Complete	athrash1
test	Complete	athrash1
test - query(file)..	Complete	athrash1
test6	Complete	athrash1

Done

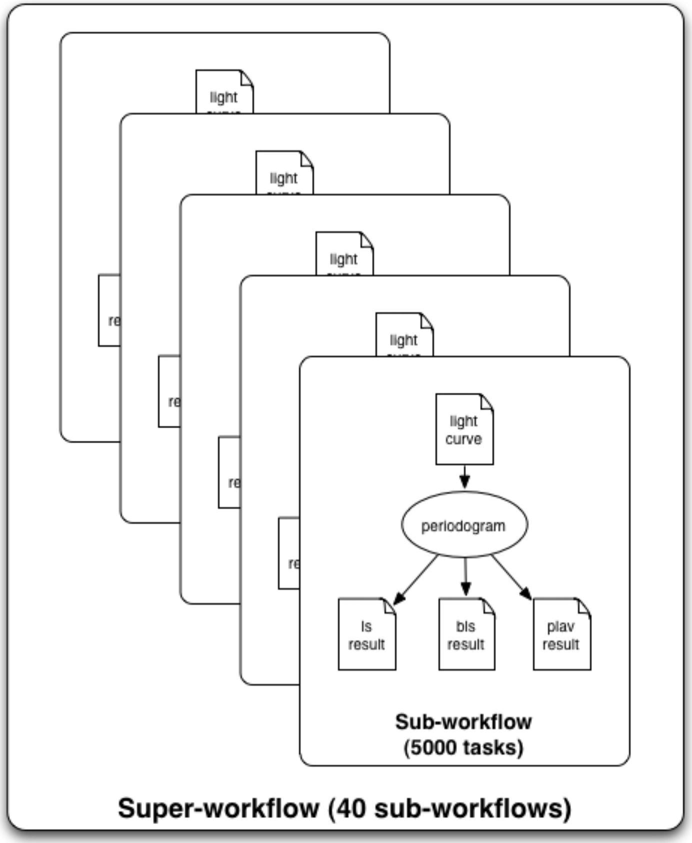






Applications in bioinformatics, molecular dynamics

Periodograms: generate an atlas of extra-solar planets

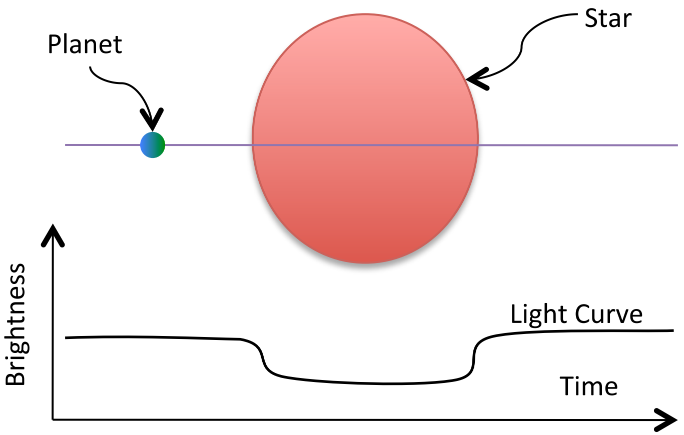


- Find extra-solar planets by
  - Wobbles in radial velocity of star, or
  - Dips in star's intensity

210k light-curves released in July 2010

Apply 3 algorithms to each curve

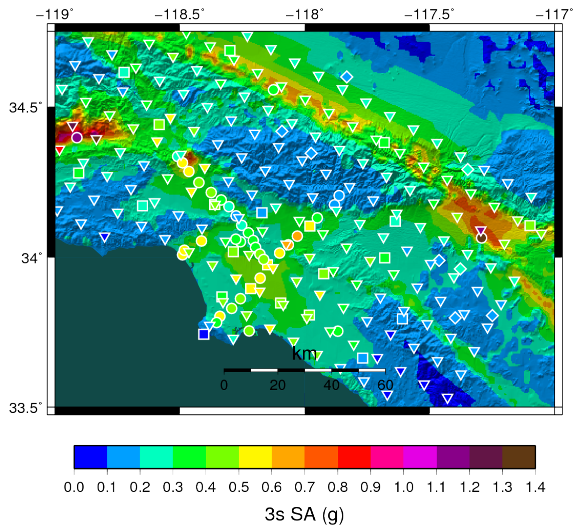
3 different parameter sets



- 210K input, 630K output files
- 1 super-workflow
- 40 sub-workflows
- ~5,000 tasks per sub-workflow
- 210K tasks total

Pegasus managed workflows

Southern California Earthquake Center



239 Workflows

- Each site in the input map corresponds to one workflow
- Each workflow has:

✧ **820,000 tasks**

MPI codes ~ 12,000 CPU hours,
Post Processing 2,000 CPU hours
Data footprint ~ 800GB

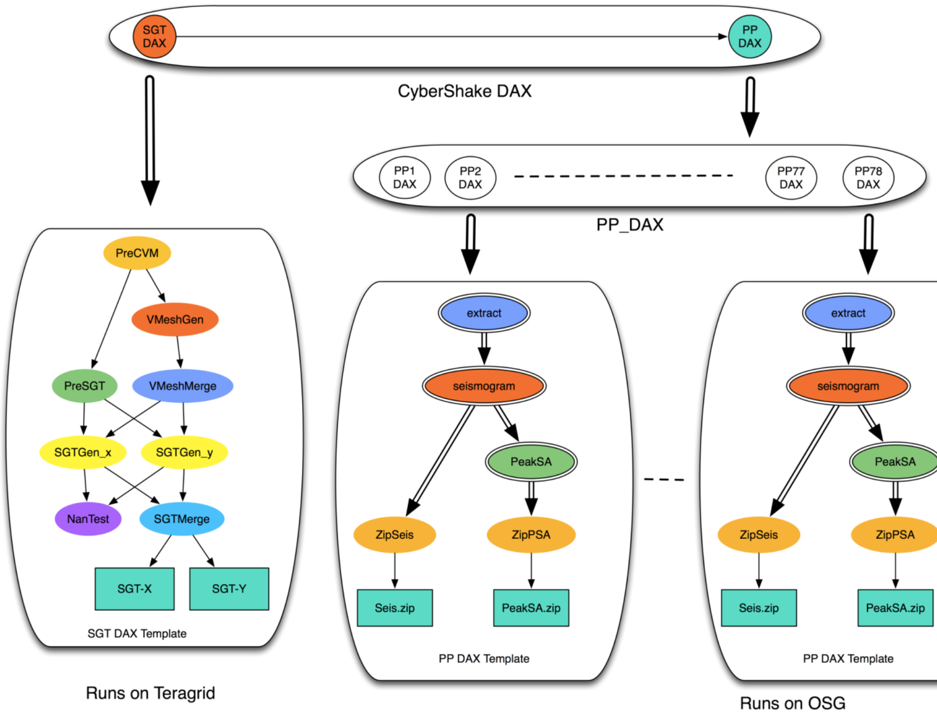
Pegasus managed workflows

CyberShake PSHA Workflow

❖ **Description**

✧ Builders ask seismologists: "What will the peak ground motion be at my new building in the next 50 years?"

✧ Seismologists answer this question using Probabilistic Seismic Hazard Analysis (PSHA)

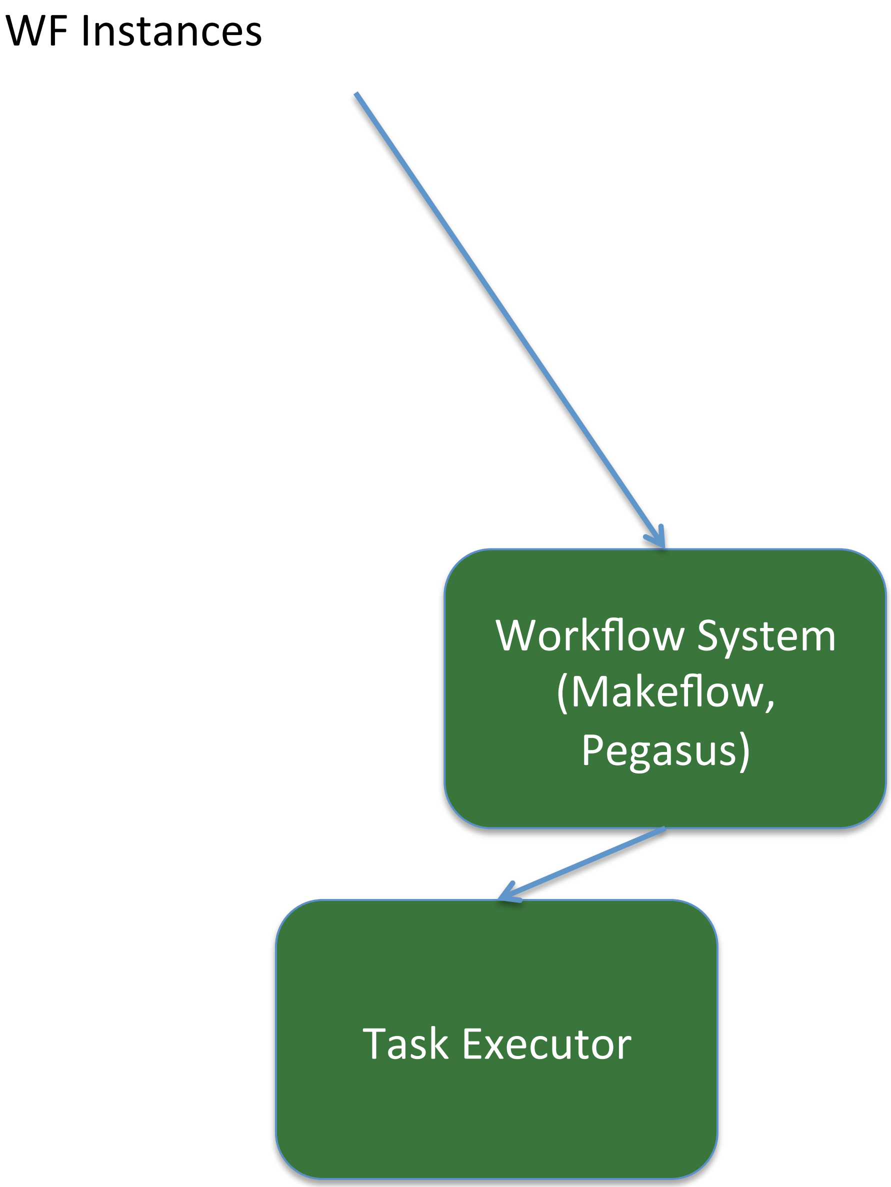


Workflow Ensembles

System Entities

- WF instance: the workflow that a user submits, has information about computations do be done and their data dependencies (WMS-specific)
- WF structure—abstract representation of the workflow (WMS-independent)
- Provisioning profile—resources needed by WF tasks (WF-independent)
- Provisioning plan– resources to be provisioned over time (WF-dependent)
- Schedule– mapping of tasks to resources (WF-dependent)

System Components

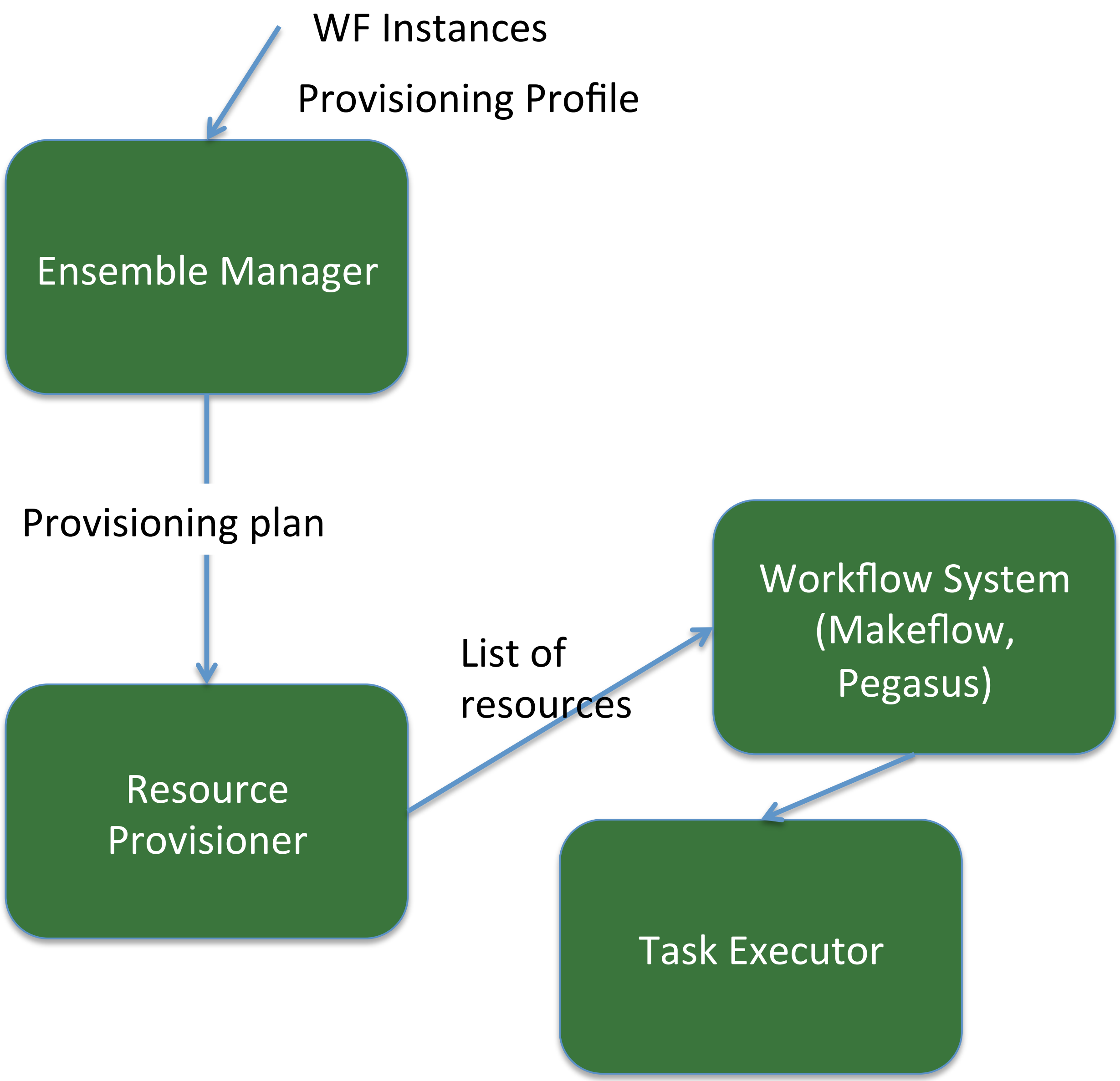


WF Instances

Workflow System (Makeflow, Pegasus)

Task Executor

System Components



WF Instances

Provisioning Profile

Ensemble Manager

Provisioning plan

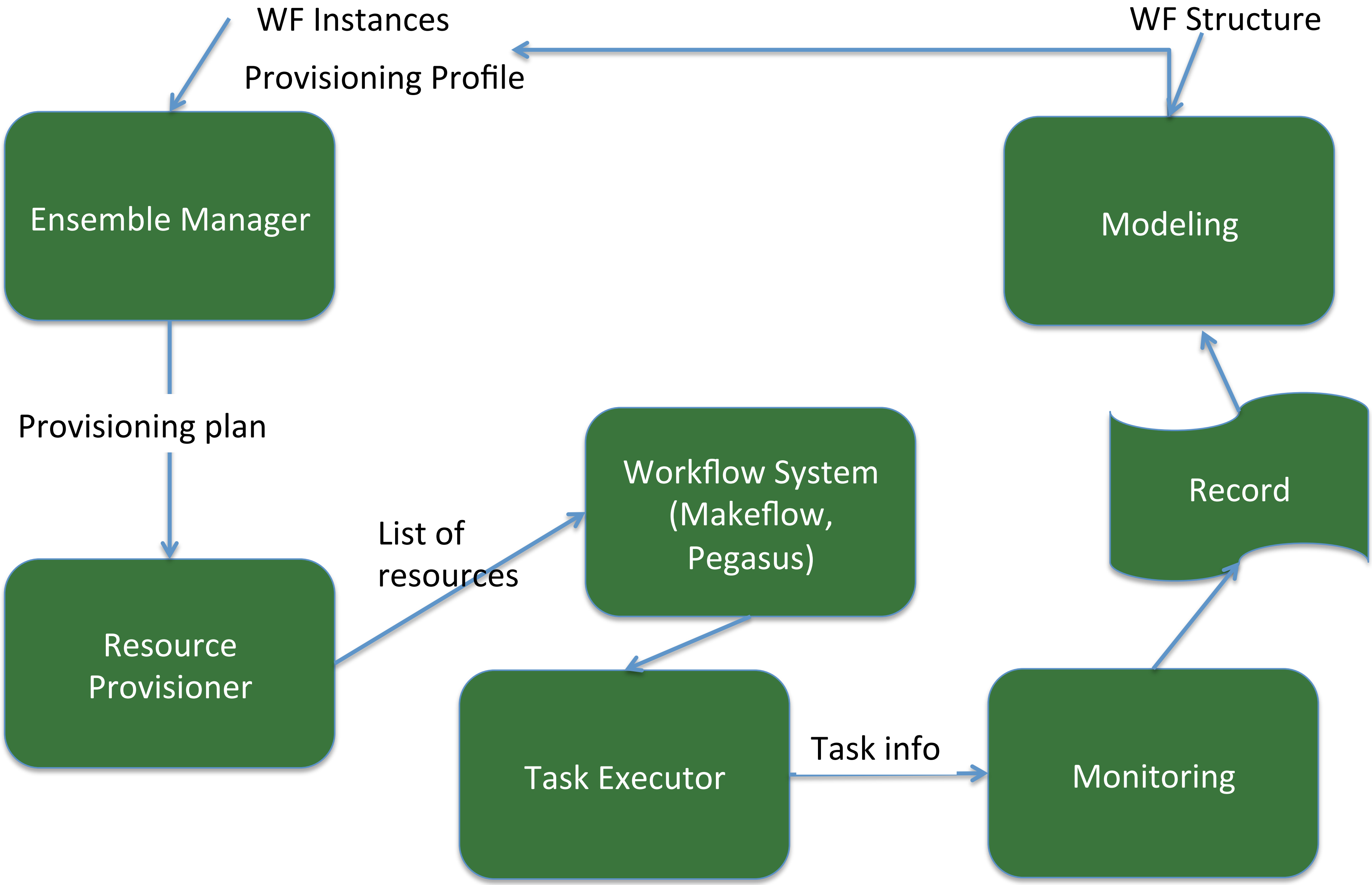
Resource Provisioner

List of resources

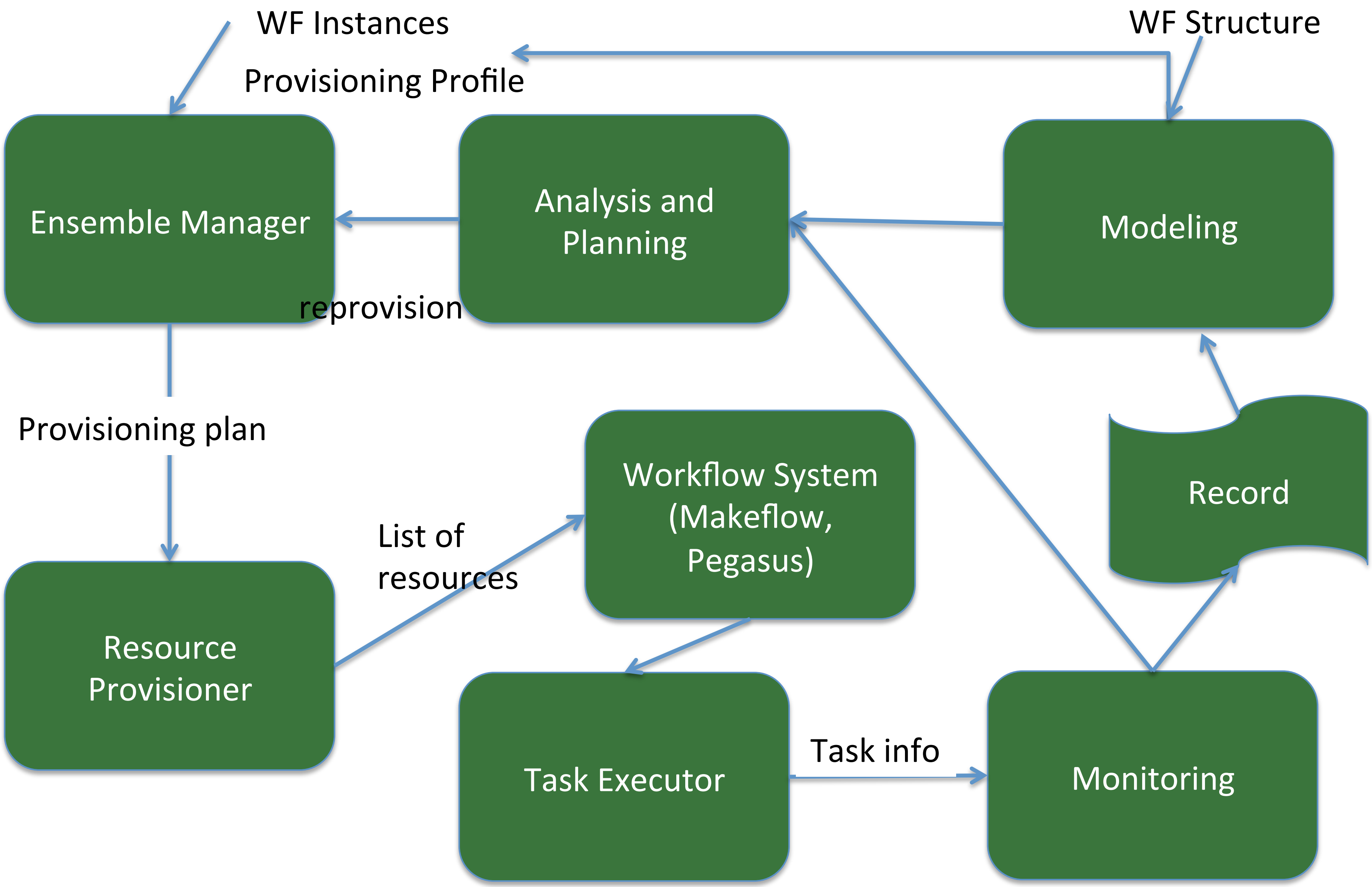
Workflow System (Makeflow, Pegasus)

Task Executor

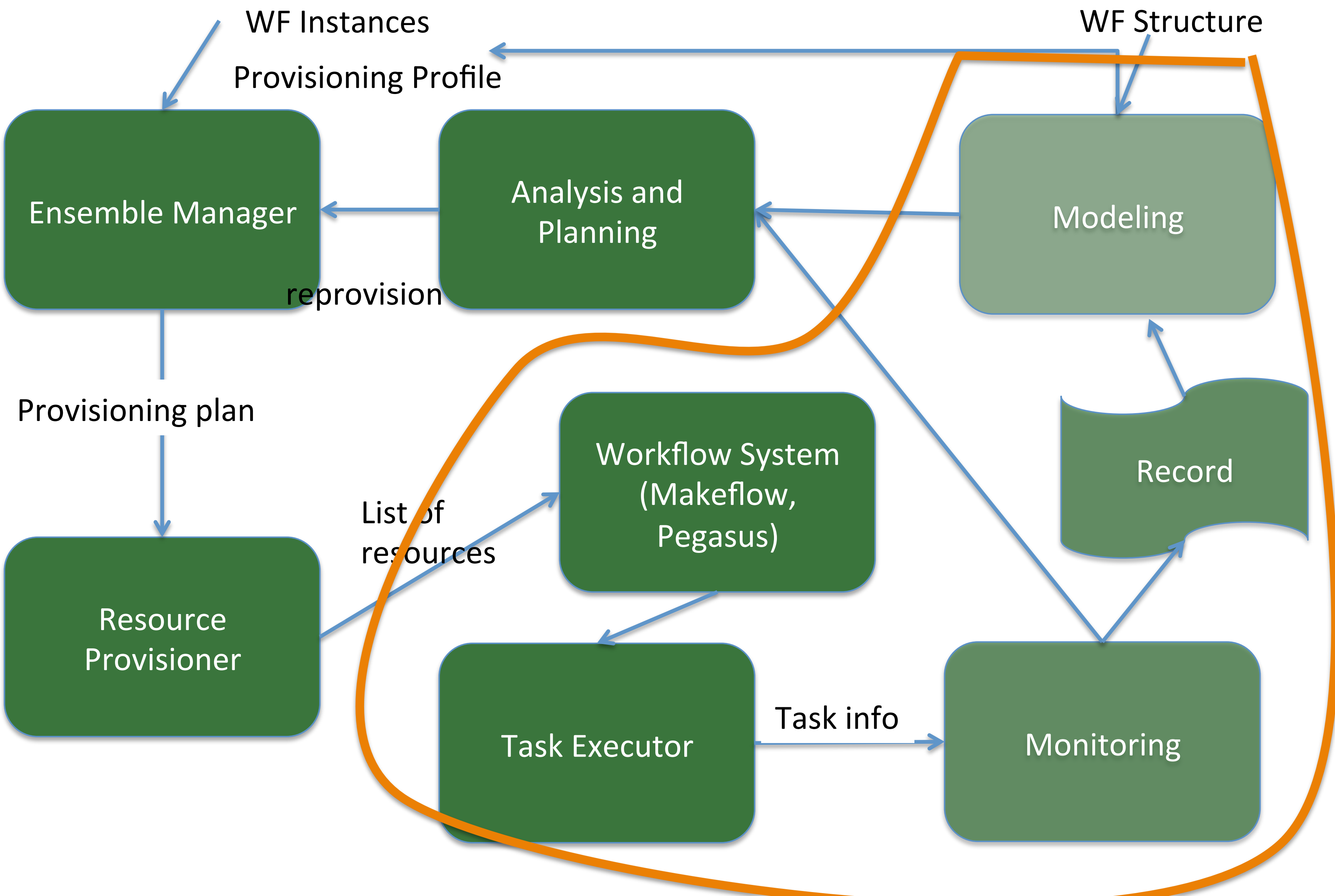
System Components



System Components



System Components



Task Characterization/Execution

- Understand the resource needs of a task
- Establish expected values and limits for task resource consumption
- Launch tasks on the correct resources
- Monitor task execution and resource consumption, interrupt tasks that reach limits
- Possibly re-launch task on different resources

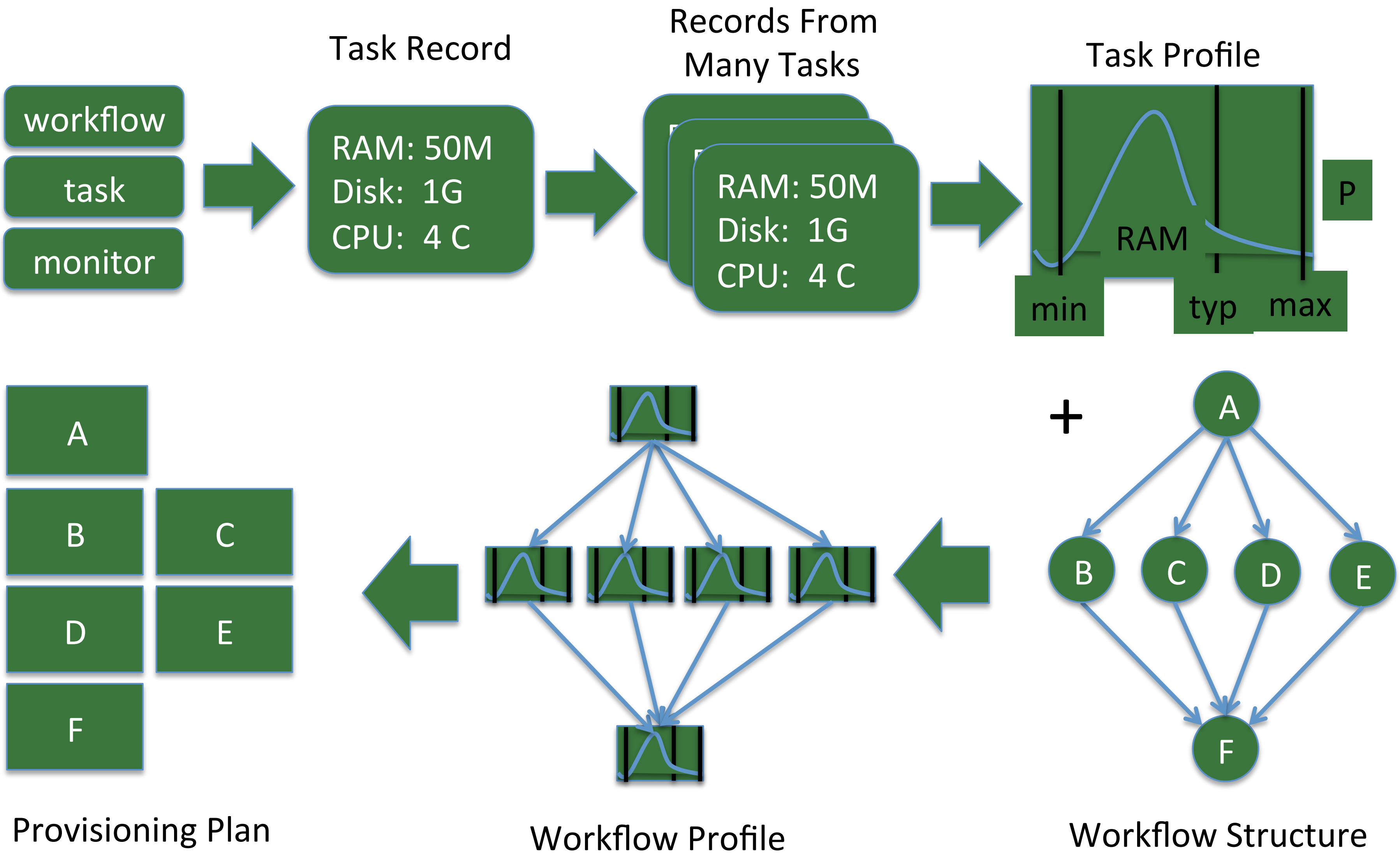
Monitoring/Modeling of tasks

- exit_type / exitcode, “signalled”, “limit”
- signal -- The number of the signal that terminated the process.
- limits_exceeded List of all the resource limits that were exceeded by the process
- max_concurrent_processes--The maximum number of processes that ran concurrently.
- cpu_time/ wall time
- peak virtual_memory/resident_memory
- bytes_read/bytes_written

Values available if a Working Directory is specified

- workdir_number_files_dirs-- The peak value of the number of files and directories in the working directory.
- workdir_footprint---The peak value of the size of all files and directories in the working directory.

Data Collection and Modeling



workflow

task

monitor

Task Record

RAM: 50M
Disk: 1G
CPU: 4 C

Records From Many Tasks

RAM: 50M
Disk: 1G
CPU: 4 C

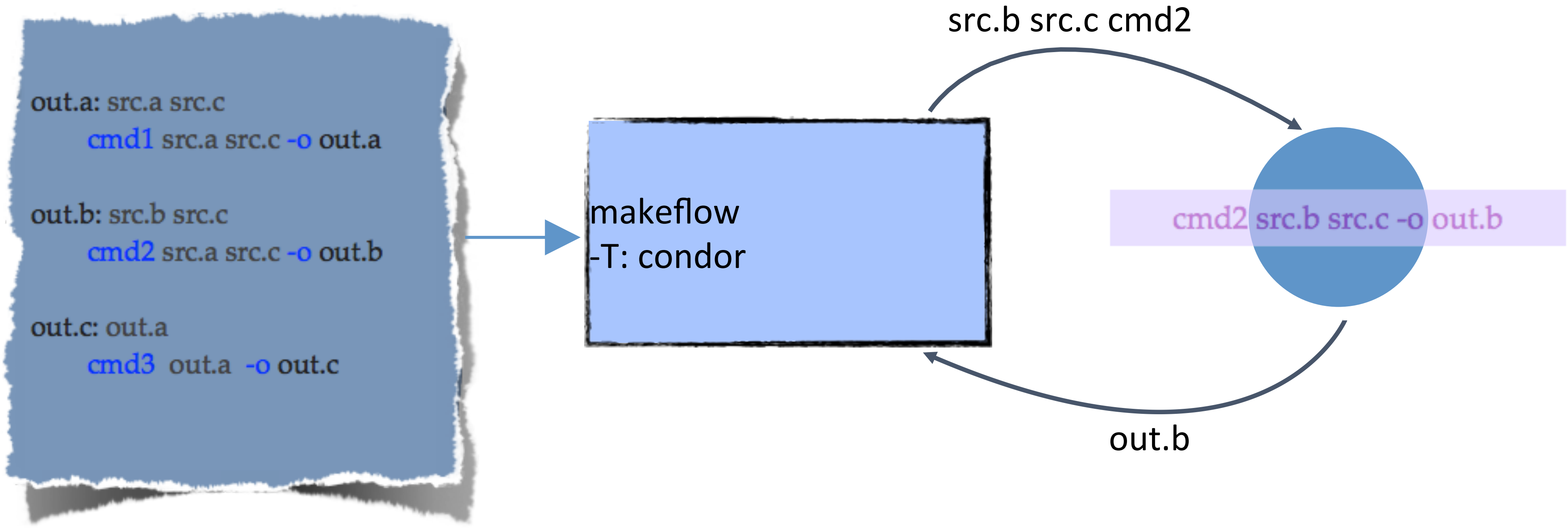
Task Profile

Workflow Structure

Workflow Profile

Provisioning Plan

Static Workflow Monitoring



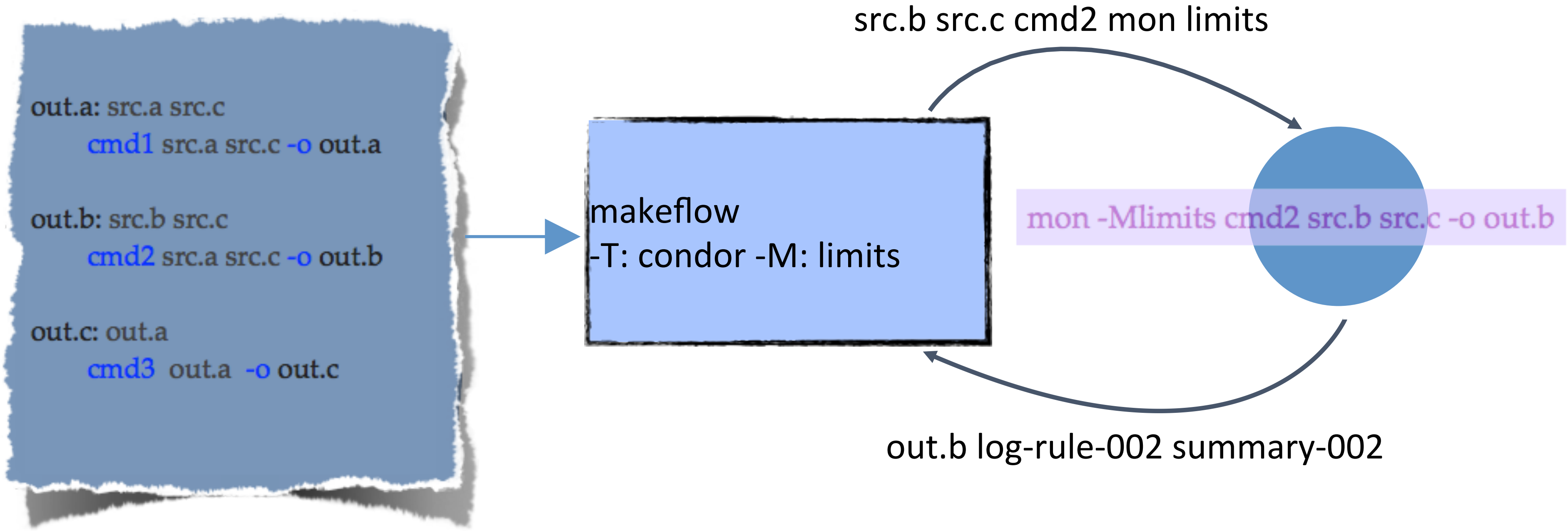
```
out.a: src.a src.c
        cmd1 src.a src.c -o out.a

out.b: src.b src.c
        cmd2 src.a src.c -o out.b

out.c: out.a
        cmd3  out.a  -o out.c
```

flow specification

resource control

flow execution

Static Workflow Monitoring



```
out.a: src.a src.c
	cmd1 src.a src.c -o out.a

out.b: src.b src.c
	cmd2 src.a src.c -o out.b

out.c: out.a
	cmd3  out.a  -o out.c
```

makefloW
-T: condor -M: limits

src.b src.c cmd2 mon limits

```
mon -Mlimits cmd2 src.b src.c -o out.b
```

out.b log-rule-002 summary-002

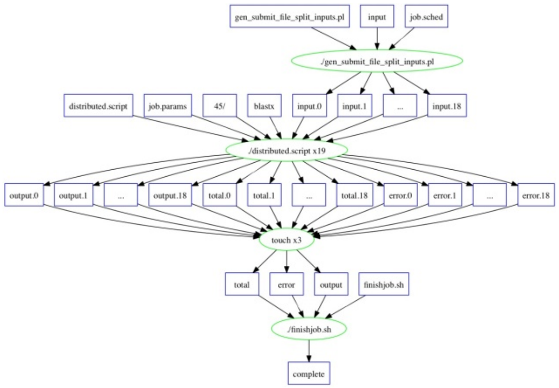
flow specification

resource control

flow execution

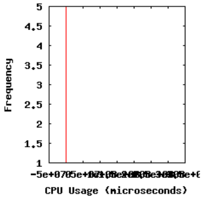
Working on making the tools generic

Portal Generated Workflows

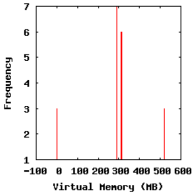


BLAST (Small)
17 sub-tasks
~4h on 17 nodes

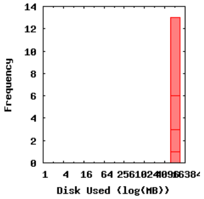
~1s
CPU

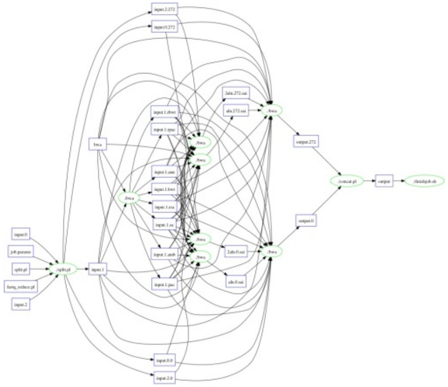


~250MB
RAM



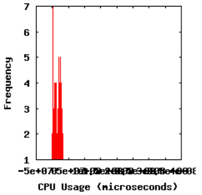
8.5GB
Disk



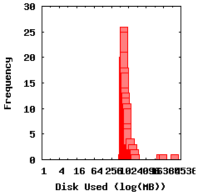


BWA
825 sub-tasks
~27m on 100 nodes

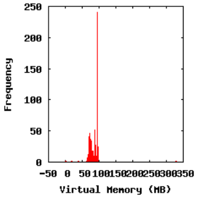
~30s
CPU

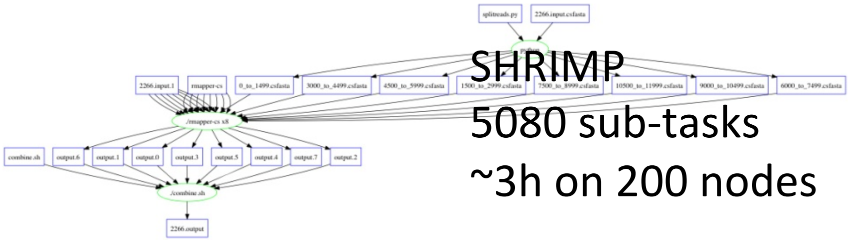


~500MB
RAM



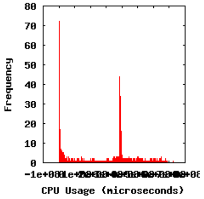
~500MB
Disk



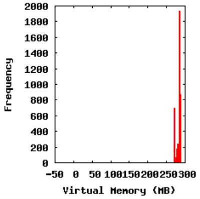


SHRIMP
5080 sub-tasks
~3h on 200 nodes

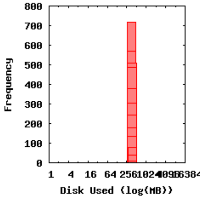
~400s
CPU



~280MB
RAM



~375MB
Disk



Experimental design

- Characterize a set of applications, run large number of instances, develop application models
- Design synthetic applications with a desired behavior (CPU consumption, Mem, I/O)
  - Run a large number of instances
  - Model task and application behavior
  - See if the model matches the input
  - See if the system responds appropriately
- Experimental platform:
  - Open Science Grid (with glideinWMS)
  - Argonne Leadership Computing Facility
  - ND/Center for Research Computing, UW, UCSD
  - Clouds

Conclusions

- dV/dt will develop a planning framework to
  - characterize and manage applications
  - provision resources/monitor execution/adapt
- Provide methodologies, algorithms, and prototype solutions
- Initial focus on application resource characterization and monitoring

- https://sites.google.com/site/acceleratingexascale/